AD-A046 081   FEDERAL COBOL COMPILER TESTING SERVICE WASHINGTON D C
              FORTRAN COMPILER VALIDATION SUMMARY REPORT.(U)
              1977
UNCLASSIFIED   FCVS66-VSR255

F/G 9/2

NL

1 OF 1
ADA046081

END
DATE
FILMED
12-77
DDC

ADA046081

④⑫
ßs

⑥ FORTRAN COMPILER
VALIDATION SUMMARY REPORT.

⑭
VALIDATION NUMBER FCVS66-VSR255

⑪ 1977

⑫ 17 p.

Date : 1977

Prepared By:

FEDERAL COBOL COMPILER TESTING SERVICE
DEPARTMENT OF THE NAVY
WASHINGTON, D.C. 20376

D D C
RECEIVED
NOV 2 1977
R

408438

FCVS66-VSR255

FORTRAN COMPILER VALIDATION
------- --------- ----------

1. Validation Number   FCVS66-VSR255

2. Vendor   Burroughs Corporation

3. Mainframe   Burroughs B7700

4. Compiler Identification   FORTRAN II.9

5. Operating System Identification   MCP II.9

6. Compiler Validation System Version Number   FCVS66 1.2

*PLEASE NOTE. The Department of the Navy may make full and free public disclosure of the Validation Summary Report (VSR) in accordance with the "Freedom of Information Act" (5 U.S.C. #552). The results of this validation are only for the purpose of satisfying United States Government requirements, and apply only to the Computer System, Operating System release, and compiler version identified in the VSR. The FORTRAN Compiler Validation System is used to determine, insofar as is practical, the degree to which the subject compiler conforms to American Standard FORTRAN, X3.9-1966. Thus, the VSR is necessarily discretionary and judgmental. The United States Government does not represent or warrant that the statements, or any one of them, set forth in the VSR are accurate or complete. The VSR is not meant to be used for the purpose of publicizing the findings summarized therein.

For information concerning this compiler you can contact the vendor's designated representative named below:

    Mr. Jay Wolf
    Burroughs Corporation
    Federal and Special Systems Group
    P.O. Box 517
    Paoli, Pennsylvania 19301

TABLE OF CONTENTS

SECTION 1.   INTRODUCTION

1.1   Purpose of the Validation Summary Report

The purpose of the Validation Summary Report (VSR) is to identify individual
FORTRAN language elements whose implementation does not conform to the language
specifications defined in American Standard FORTRAN, X3.9-1966.

1.2   Preparation of the VSR

The Validation Summary Report is prepared by analyzing the results of running
the FORTRAN Compiler Validation System (FCVS). The FORTRAN Compiler Validation
System consists of audit routines containing features of American Standard
FORTRAN, their related data, and an Executive Routine which prepares the
audit routines for compilation. Each audit routine is a FORTRAN program which
includes many tests and supporting procedures indicating the result of the
tests.

The testing of a compiler in a particular hardware/operating system environment
is accomplished by compiling and executing each audit routine. The report pro-
duced by each routine tells whether the compiler passed or failed the tests in
the routine. If the compiler rejects some language elements by terminating
compilation, giving fatal diagnostic messages, or terminating execution abnor-
mally, then the test containing the code the compiler was unable to process is
deleted. The audit routine is compiled again and execution is repeated.

The compilation listings and the output reports of the audit routines con-
stitute the raw data from which the members of the Federal COBOL Compiler
Testing Service produce a Validation Summary Report.

1.3   Organization of the VSR

The Validation Summary Report is made up of several sections whose contents
are described below.

     a.   Section 2 summarizes the results of the compilation and execution
of the programs comprising the FORTRAN Compiler Validation System. Section 2
is divided into a subsection describing the syntax errors encountered while
compiling the FORTRAN audit routines, and a subsection describing the semantic
errors which occurred during execution of the FORTRAN audit routines.

     b.   Section 3 contains information which describes the software environ-
ment in which the compiler was tested. This includes the name and version of
the operating system and the logical unit/physical device assignments used in
the programs comprising the FCVS. The options used with the compiler are also
given, and if applicable, the use of compiler optimization features is ex-
plained.

     c.   Appendix A is the Validation Summary Working Document, a working
paper resulting from the compilation and execution of the FCVS. The VSR
is derived from Appendix A.

1.4.   Use of the VSR

The Department of the Navy may make full and free public disclosure of the Validation Summary Report (VSR) in accordance with the "Freedom of Information Act" (5 U.S.C. #552). The results of the validation are only for the purposes of satisfying United States Government requirements, and apply only to the computer system, operating system release, and compiler version identified in the VSR.

The FORTRAN Compiler Validation System is used to determine, insofar as is practical, the degree to which the subject compiler conforms to the FORTRAN Standard. Thus, the VSR is necessarily discretionary and judgmental. The United States Government does not represent or warrant that the statements, or any one of them, set forth in the VSR are accurate or complete. The VSR is not meant to be used for the purpose of publicizing the findings summarized therein.

1.5 Sources of Additional Information

The detailed FORTRAN language specifications are given in the publication "American Standard FORTRAN, X3.9-1966", available from the American National Standards Institute, 1430 Broadway, New York, New York 10018.

An explanation of the FORTRAN Compiler Validation System is contained in the FCVS User's Guide. This document explains how to run the compiler validation system. The User's Guide and a magnetic tape containing a copy of the FCVS programs are available from the National Technical Information Service, 5285 Port Royal Road, Springfield, Virginia, 22151. (Ordering information can be obtained from the Federal COBOL Compiler Testing Service.)

SECTION 2.   DETAILED EVALUATION OF ERRORS.

This section summarizes the results of the compilation and execution of the programs comprising the FORTRAN Compiler Validation System (FCVS). The version of the FCVS used during this validation is shown inside the front cover of the VSR.

Section 2 is made up of two subsections. The first subsection describes each syntax error encountered during compilation of the audit routines, and the second subsection describes the semantic errors encountered during execution of the audit routines.

Each error or deviation noted in this section makes reference to a program contained in Appendix A (Validation Summary Working Document). This reference provides the documented results of an occurrence of errors/deviations detected during the running of the FCVS using the compiler within the environment identified in this document. The Validation Summary Working Document is presented in sequence by program number.

2.1   Syntactical Errors

The compiler produced a fatal error message for an arithmetic assignment statement containing 57 nested parentheses. (See FM045.)

X3.9-1966 References: Section 6.1, Arithmetic Expressions
                       Section 7.1.1.1, Arithmetic Assignment Statement

2.2   Semantic Errors

No semantic errors occurred during the execution of the FCVS audit routines.

SECTION 3. SOFTWARE ENVIRONMENT.

The compiler referenced in this document was validated using the software environment described in this section. When using a modification of the described environment, the compiler may or may not continue to conform to the Standard. It should be noted that during the validation process, an attempt is made to validate as many different options as possible.

The use of compiler options, logical unit/physical device assignments, and any form of optimization which is not described in this report could cause the compiler to produce a program that does not perform according to the specifications of Standard FORTRAN. Only the environment described in this document has been used with this compiler to satisfy the validation require- ments of the Department of the Navy.

1. Options or parameters used on the processor call statement for the compiler.

Options specified:

```
? COMPILE FMnnn [ET] WITH FORTRAN [CT];
DATA CARD
$ SET LINEINFO
```

2. Logical Unit/Physical Device Assignments.

Printer Destined Files:

Logical unit 06 was assigned to a printer output file by using the following FILE source card within the program:

```
FILE  6(KIND=6,MAXRECSIZE=22)
```

and ET control card:

```
? ET(FILE FILE6(TITLE=FCVS/FILE6,INTMODE=EBCDIC));
```

Tape Files:

Logical unit 07 was assigned to a tape file by using the following FILE source card within the program:

```
FILE  7(KIND=15,MAXRECSIZE=20,BLOCKSIZE=300)
```

and ET control card:

```
? ET(FILE FILE7(TITLE=FCVS/FILE7,INTMODE=EBCDIC,AREAS=100,
           AREASIZE=90,SAVEFACTOR=2,PROTECTION=SAVE));
```

Sequential Mass-storage Files:

Logical unit 07 was assigned to a mass-storage device by using the following FILE source card within the program:

FILE 7(KIND=1,MAXRECSIZE=20,BLOCKSIZE=300)

and the ET control card:

? ET(FILE FILE7(TITLE=FCVS/FILE7,INTMODE=EBCDIC,AREAS=100,
              AREASIZE=90,PROTECTION=SAVE));

Card Input Files:

Logical unit 05 was assigned to a mass-storage file which contained
the card image read by FM015. The FILE source card used was:

FILE 5(KIND=1,FILETYPE=7)

and the ET control card was:

? ET(FI   FILE5(TITLE=FCVS/FILE5,INTMODE=EBCDIC));

3. Optimization   compiler may or may not have optimization features.
If there was an   imization feature available, it was used during the
validation process (during a separate execution of the Compiler Validation
System) to determine if its use causes the compiler to produce a program which
does not give the expected results. If the optimization is invoked through
the compiler call statement then it is mentioned in paragraph 1 above. If
it is invoked through the introduction of a compiler directing source program
statement, it is shown below. Optimization which would require modification
to source program statments is not considered in this report in that it is
beyond the scope of the use of Standard FORTRAN and the validation process.

The default optimization setting (OPT=0) was used for this validation. The
The compiler was not validated for any other optimization settings.

4. Compiler.

FORTRAN II.9

5. Operating system.

MCP II.9

APPENDIX A

VALIDATION SUMMARY WORKING DOCUMENT

This appendix is a working paper produced during the validation and documents the results of the compilation and execution of each of the programs comprising the FCVS. The results contained herein are based on the use of the compiler within the Validation Environment identified in this appendix. This appendix (Validation Summary Working Document) is not part of the official Validation Summary Report (VSR), and it is not intended to reflect in any way the compiler's usefulness or degree of conformance to the language specifications.

The reader of this appendix should keep in mind that the same problem area may appear in more than one program but is considered only as a single discrepancy, and the problem is reflected only once in the body of the VSR. (The VSR will in turn only reference the first occurrence of the problem in the appendix.)

This appendix is divided into four parts. The first part describes the Validation Environment. The second part lists the Monitor Input Cards used in creating a job control stream for execution in the batch mode. The third part shows the control cards required to compile and execute an individual program. The fourth part of the document is divided into two categories of information: compilation results and execution results. Information items, such as compiler warning messages, are included in the summary of compilation and execution results.

The reference document for FORTRAN is American Standard FORTRAN, X3.9-1966.

VALIDATION ENVIRONMENT

COMPILER IDENTIFICATION: FORTRAN II.9

COMPUTER SYSTEM: Burroughs B7700

OPERATING SYSTEM: MCP II.9

    Each of the programs comprising the FCVS was compiled with the default
optimization setting (OPT=0). The programs which test I/O, FM100 through
FM108, were run twice. The programs were first run with the output logical
unit assigned to a mass-storage device, and then rerun with the output
logical unit assigned to a tape device.

FCVS66-VSR255

CCVS MONITOR INPUT CARDS

The CCVS Executive Routine was used to prepare the FCVS programs for
execution. The Monitor Control Cards used as input to the CCVS Executive
are listed below:

```
          *CCVSVR NONE
          *DATE 770913
          *ALTSYSCODE *
          *LIST UPDATES,XCARDS,CTL,INSERT

             (FORTRAN routine selection cards)

   I-01     * JOB VALIDATEFORTRAN;
   I-02     USER=XXXXXXXXXXX
   I-03     FAMILY DISK=LSSPACK OTHERWISE PACK;
   I-04     BEGIN
   I-05     * ET(FILE FILE5(TITLE=FCVS/FILE5,INTMODE=EBCDIC));
   I-06     * ET(FILE FILE6(TITLE=FCVS/FILE6,INTMODE=EBCDIC));
   I-07     * ET(FILE FILE7(TITLE=FCVS/FILE7,INTMODE=EBCDIC,AREAS=100,
   I-08                     AREASIZE=90,PROTECTION=SAVE));
   I-09
   B-0111   * COMPILE XXXXX [ET] WITH FORTRAN [CT];
   B-02     DATA CARD
   B-03     $ SET LINEINFO
   B-04     FILE  5(KIND=1,FILETYPE=7)
   B-05     FILE  6(KIND=6,MAXRECSIZE=22)
   B-06     FILE  7(KIND=1,MAXRECSIZE=20,BLOCKSIZE=300)
   B-07
   E-01
   T-01     END JOB
   T-02
          *END-MONITOR
```

FCVS66-VSR255

CONTROL CARDS FOR RUNNING FCVS PROGRAMS

The job control stream for running the FCVS programs consisted of the
following control cards:

```
? JOB VALIDATEFORTRAN;
USER=XXXXXXXXXXX
FAMILY DISK=LSSPACK OTHERWISE PACK;
BEGIN
? ET(FILE FILE5(TITLE=FCVS/FILE5,INTMODE=EBCDIC));
? ET(FILE FILE6(TITLE=FCVS/FILE6,INTMODE=EBCDIC));
? ET(FILE FILE7(TITLE=FCVS/FILE7,INTMODE=EBCDIC,AREAS=100,
             AREASIZE=90,PROTECTION=SAVE));
? COMPILE FM001 [ET] WITH FORTRAN [CT];
DATA CARD
$ SET LINEINFO
FILE  5(KIND=1,FILETYPE=7)
FILE  6(KIND=6,MAXRECSIZE=22)
FILE  7(KIND=1,MAXRECSIZE=20,BLOCKSIZE=300)

   (FORTRAN source code for program FM001)

? COMPILE FM002 [ET] WITH FORTRAN [CT];
DATA CARD
$ SET LINEINFO
FILE  5(KIND=1,FILETYPE=7)
FILE  6(KIND=6,MAXRECSIZE=22)
FILE  7(KIND=1,MAXRECSIZE=20,BLOCKSIZE=300)

  (FORTRAN source code for program FM002)


        .
        .
        .


END JOB
```

RUN SUMMARIES

FM001 through FM012

    A. Compilation

       No errors.

    B. Execution

       No errors.

FM013

    A. Compilation

       The statement

            GO TO I, (1262,1263,1264)

       where neither 1262 nor 1264 appear in an ASSIGN statement
       generates the warning messages

       WARNING: "1262" LABEL APPEARED IN ASSIGNED-GO-TO BUT NOT ASSIGN
       WARNING: "1264" LABEL APPEARED IN ASSIGNED-GO-TO BUT NOT ASSIGN

       These are valid warning messages.

    B. Execution

       No errors.

FM014

    A. Compilation

       No errors.

    B. Execution

       No errors.

FM015

    A. Compilation

       No errors.

    B. Execution

1. The PAUSE statement, PAUSE 0123, displays the lines

        DISPLAY:PAUSE 123.
        PROGRAMATICALLY SUSPENDED

2. The octal digits in a STOP 0247 statement are not displayed.

The above results are not considered errors and are included for completeness only. For the PAUSE statement the FORTRAN Standard states, "At the time of cessation of execution the octal digit string is accessible."

In a published clarification, the FORTRAN Standard Committee stated: "The accessibility of n in a STOP statement is intentionally not specified in the standard. By not so specifying, the standard permits the practice of terminating program execution without necessarily making n accessible."

FM016 through FM044

A. Compilation

    No errors.

B. Execution

    No errors.

FM045

A. Compilation

    Test 759 in FM045 was flagged as a fatal error. The statement tests the use of 57 nested parentheses to enclose the right-hand side of an arithmetic assignment statement. The source code is:

        IVCOMP = (((
                 IVON01 / IVON02
                 )))

    The fatal message stated:

      "(" COMPILER LIMIT FOR OPT=0 EXPRESSIONS EXCEEDED--DIVIDE THE
          EXPRESSION INTO SUBEXPRESSIONS

    (Although not part of the official validation which was performed for the default optimization setting (OPT=0), the program FM045 compiled successfully with OPT=1 specified.)

B. Execution

    Test 759 had to be deleted. All other tests executed correctly.

FM050 through FM062

A. Compilation

No errors.

B. Execution

No errors.

FM080 through FM083

A. Compilation

No errors.

B. Execution

No errors.

FM097 through FM099

A. Compilation

No errors.

B. Execution

No errors.

FORTRAN I/O Programs - FM100 through FM109

The I/O programs were executed with the output logical unit assigned to a tape device and rerun with the output logical unit assigned to a mass-storage device.

FM100 through FM109

A. Compilation

No errors.

B. Execution

No errors.

BIBLIOGRAPHIC DATA SHEET	1. Report No. FCVS66-VSR255	2.	3. Recipient's Accession No.

4. Title and Subtitle	5. Report Date
Validation Summary Report #FCVS66-VSR255 Burroughs B7700 FORTRAN II-9	
	6.

7. Author(s) Same as organization - see 9.	8. Performing Organization Rept. No.

9. Performing Organization Name and Address Federal COBOL Compiler Testing Service Department of the Navy Washington, DC 20376	10. Project/Task/Work Unit No.
	11. Contract/Grant No.

12. Sponsoring Organization Name and Address Automatic Data Processing Equipment Selection Office Department of the Navy Washington, DC 20376	13. Type of Report & Period Covered
	14.

15. Supplementary Notes

16. Abstracts

This Validation Summary Report (VSR) for the Burroughs B7700 FORTRAN Compiler Version II.9 (MCP Version II.9) provides a consolidated summary of the results obtained from the validation of the subject compiler against the FORTRAN Standard (X3.9-1966). The VSR is made up of several sections showing the discrepancies found. These include an overview of the validation which lists all categories of discrepancies by chapter within X3.9-1966.

17. Key Words and Document Analysis.   17a. Descriptors

Programming Languages
Standards
Compilers
FORTRAN
Verifying
Proving Program Correctness
Software Engineering

17b. Identifiers/Open-Ended Terms

FCVS
CVS

17c. COSATI Field/Group   09/02

18. Availability Statement		19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages
Release unlimited.		20. Security Class (This Page) UNCLASSIFIED	22. Price